Fast Naïve Bayes classifiers for COVID-19 news in social networks

Hasan Dwi Cahyono[1], Atara Mahadewa[2], Ardhi Wijayanto[2], Dewi Wisnu Wardani[1], Haryono Setiadi[2]
[1]Department of Data Science, Faculty of Information Technology and Data Science, Universitas Sebelas Maret, Surakarta, Indonesia
[2]Department of Informatics, Faculty of Information Technology and Data Science, Universitas Sebelas Maret, Surakarta, Indonesia

Article Info

Article history:

Received Nov 5, 2023
Revised Feb 9, 2024
Accepted Feb 16, 2024

Keywords:

Fake news detection
Fast and reliable
Naïve Bayes classifiers
Real-time processing
Social media

ABSTRACT

The growth of fake news has emerged as a substantial societal concern, particularly in the context of the COVID-19 pandemic. Fake news can lead to unwarranted panic, misinformed decisions, and a general state of confusion among the public. Existing methods to detect and filter out fake news have accuracy, speed, and data distribution limitations. This study explores a fast and reliable approach based on Naïve Bayes algorithms for fake news detection on COVID-19 news in social networks. The study used a dataset of 10,700 tweets and applied text pre-processing, term-weighting, document frequency thresholding (DFT), and synthetic minority oversampling techniques (SMOTE) to prepare the data for classification. The study assessed the performance and runtime of four models: gradient boosting (GDBT), decision tree (DT), multinomial Naïve Bayes (MNB), and complement Naïve Bayes (CNB). The testing results showed that the CNB model reaches the highest accuracy, precision, recall, and F1-score of approximately 92% each, with the shortest runtime of 0.55 seconds. This study highlights the potential of the CNB model as an effective tool for detecting online fake news about COVID-19, given its superior performance and rapid processing time.

Corresponding Author:

Hasan Dwi Cahyono
Department of Data Science, Faculty of Information Technology and Data Science
Universitas Sebelas Maret
Surakarta, Indonesia
Email: hasan.dwi.cahyono@gmail.com

1. INTRODUCTION

The COVID-19 outbreak has brought an unparalleled wave to the world, not just in terms of education [1] and health [2] but also socially and economically [3]. The influence of COVID-19 has made it increasingly difficult to distinguish between real news and fake ones (hoaxes) due to the quantity of online information. Hoaxes, rumours, and conspiracy theories about COVID-19 have spread rapidly, causing uncertainty and concern among the public [4]. Detecting such concerning information during illness outbreak management (infodemic) has become crucial to prevent viral propagation and its detrimental impact on society [5]. Despite the decreasing range of international COVID-19 occurrences, the infodemic remains critical for monitoring and detecting hoaxes in real-time. As the COVID-19 case continues, new virus variants may emerge at any time. Further, the unconfirmed information about the COVID-19 vaccine and its usefulness is spreading, leading to vaccine hesitancy among the public [6]. Real-time hoax detection for COVID-19 cases can be challenging, especially with the constant influx of new information. While several algorithms are available for hoax detection, many approaches have gaps in detecting hoaxes in real-time. For instance, some algorithms rely on human intervention to train the model and detect hoaxes, making those methods time-consuming and inadequate [7]. Other algorithms may not be suitable as they were designed for non-COVID-19 cases [8].

Multinomial Naïve Bayes (MNB) is a popular nonparametric algorithm based on probability concepts used in text categorisation [9]. Compared to different classification algorithms, MNB can straightforwardly update the model when new data is added. When the assumption of the feature set is independent, Naïve Bayes classifiers are simple and yet have been indicated to have higher stability compared to other algorithms, such as maximum entropy, Bayesian networks (BN), support vector machines (SVM), and neural networks (NN) [10]. Regarding misclassification costs, a different study provides evidence from a range of findings that show that MNB outperforms different algorithms, such as SVM and logistic regression (LR) [11]. According to this study, MNB is considered a suitable algorithm for real-time hoax detection since MNB has low computational complexity, higher accuracy, and is faster than other considered methods. Using MNB to classify harmful websites, a different study reached an average precision of up to 91.30% [12]. To identify fraudulent activities in US SEC registration data, an investigation also found that MNB had the highest misclassification costs compared to ensemble approaches, LR, decision tree (DT), SVM, and NN [11]. These prior findings from different studies indicated that the Naïve Bayes technique is a viable option for real-time hoax detection applications due to its robust performance and low computational complexity. A recent study explored fake news classification using COVID-19 data in English [13]. However, those classification algorithms were evaluated in accuracy, precision, recall, and F1-scores, making the computational complexity unclear. Additionally, those evaluations were performed without considering hyperparameter tuning to enhance performance.

Thus, this study investigates Naïve Bayes classifiers to potentially fill the gaps in real-time fake detection while providing fast and reliable algorithms based on an investigation framework for COVID-19 cases [13]. Nevertheless, Naïve Bayes classifiers such as MNB are highly dependent on assumptions to achieve peak performance and highly susceptible to overfitting, mainly when the dataset is imbalanced. To mitigate the assumption dependencies, a modified MNB known as complement Naïve Bayes (CNB) shows superior performance over and above the classical MNB. Further, integrating an oversampling technique based on synthetic minority oversampling techniques (SMOTE) could potentially balance the distribution of the dataset. To analyse the reliability and computational complexity, both MNB and CNB performance metrics and runtime will be measured with and without the integration of SMOTE. To the best of our knowledge, these investigations are novel.

2. RELATED WORKS

Social media platforms have led to the rapid spread of hoaxes, which can cause social unrest and misinformation. Detecting hoaxes efficiently and effectively is crucial to prevent their spreading. This literature review examines the real-time cases of hoax detection and the importance of machine learning algorithms for hoax detection.

2.1. Real-time hoax detection in social media networks

Real-time hoax detection can effectively identify and debunk fake information, mitigating its influence, irrespective of the subject matter. Hoaxes on social events [14], natural catastrophes [15], elections [16], and the COVID-19 outbreak [17] have the possibility to inflict harm and disseminate misinformation, thereby fostering social discontent, eroding trust in institutions, and posing threats to public health. Social gatherings, such as protests, are susceptible to manipulation by those who disseminate misinformation to influence public sentiment and instigate acts of aggression [16]. In a comparable vein, the dissemination of inaccurate data concerning natural calamities has the potential to stimulate confusion and alarm. Likewise, the propagation of deceptive claims about electoral proceedings can erode confidence in democratic systems.

The COVID-19 outbreak has underscored the need for real-time fake identification on social media platforms. Although distinctions exist between hoaxes relating to COVID-19 and those unrelated to the virus, there are also parallels in the imperative for timely detection of such deceptive information. Specific hoaxes can overlap in both domains, as exemplified by unclarified presumptions about the origin and spreading of the COVID-19 virus. The application of machine learning algorithms has significantly enhanced the ability to detect hoaxes in real-time, facilitating prompt identification and response to misleading material. Nevertheless, the pressing need to tackle frauds connected to COVID-19 highlights the necessity of ongoing study and advancement in this domain [18].

2.2. Machine learning algorithms for hoax classification

Machine learning algorithms detect data patterns to recognise infectious diseases [19]. Further, those algorithms can be trained using an extensive dataset of true and false information to develop an accurate and reliable hoax detection system. Among these algorithms, MNB has shown high accuracy in identifying hoaxes in social networks. In general, MNB is a probabilistic-driven algorithm for classification that measures the likelihood of a tweet belonging to a particular category based on the total of words within the tweet [6], [20]. Several investigations have compared the performance of MNB with other classification algorithms for hoax detection. For example, a study compared the performance of MNB, k-nearest neighbour (k-NN), and DT users of Twitter [21]. Even when the data being investigated were extensively limited, the study found that MNB can perform substantially superior to other algorithms in recall, despite MNB having significantly lower computational complexity. Recent studies found that MNB outperformed SVM, LR, and random forest (RF) for spam detection [22]. Different research also shows that MNB is considerably more promising at analysing sentiment than k-NN [23], the classic Naïve Bayes, and Naïve Bayes with Bernoulli approach (BNB) [9].

Further, computation speed is critical in selecting a classification algorithm. A study [24] compared the MNB-based classifier with the more powerful approach of recurrent neural networks (RNN) and convolutional neural networks (CNN) for detecting fake news. They discovered that the original MNB, followed by RF, achieved superior outcomes measured in accuracy, precision, recall, and F1-score. This accelerated computation makes MNB-based methods a suitable algorithm for real-time hoax detection in social networks. Continued research and development for accurate and real-time algorithms is of interest to address the growing concern about hoaxes in social networks.

3. METHODS

Preventing the spread of hoaxes and misinformation is critical for maintaining a safe and informed online community. Using only the provided datasets and source codes, we follow the investigation framework [13] for fair and comparable results for gradient boosting (GDBT) and LR. Subsequently, we modified the original source code to accommodate our explorations of Naïve Bayes algorithms and SMOTE. In our investigation, text processing comprises essential steps for robust structured data addressing high dimensionality. Classification algorithms improve online users' safety and awareness in social networks by selecting the best hyperparameters while considering imbalanced datasets to improve classification performance. Further, we evaluate the performance and runtime of classifiers. The source code of this research can be found at uns.id/fake_news_covid19.

3.1. Tweet data and text pre-processing

For a comparable investigation, the tweet data for this investigation were obtained from a study [13] where the news dataset used came from popular social networking media, explicitly, Facebook, and Twitter, as well as several fact-checking websites, namely PolitiFact, NewsChecker, and Boomlive. The total tweet data used consists of 10,700 and is grouped into two classes: real and fake, which are divided for the training, validation, and testing processes. The dataset comprised 6,420 training data, 2,140 validation data, and 2,140 testing data. The distribution of the real and fake labels in training data can be seen in Figure 1.

Text pre-processing is essential in converting unstructured textual data into structured data, which is then stored in a database [11]. The primary aim of text pre-processing is transforming the unprocessed text (raw) input into a structure that machine learning algorithms can effectively interpret and evaluate. In this study, the text pre-processing approach involved several stages, starting with case folding, eliminating all non-letter characters, such as numbers and punctuation marks, and converting all capital letters to lowercase. This textual processing step was necessary to ensure seamless data processing and querying. The next step involved tokenisation, in which the folded characters were segmented into individual words, forming a set of unique tokens that could be used to prepare a vocabulary for the corpus. Finally, the indescribable words (stop words) that can be discarded in the bag-of-words approach, were removed to reduce the number of words. Thus, performing all these steps will eventually optimise the algorithm of the classification process. These text pre-processing steps (Figure 2) are critical in ensuring the reliability and validity of the research findings.

3.2. Data representation using term-weighting

To represent the formatted data, first, we use an extracted k set of terms $t = \{t_1, \ldots, t_k\}$ from the m number of posts $d = \{d_1, \ldots, d_m\}$ and their corresponding labels $y = \{y_1, \ldots, y_m\}$ where $y \in \{\text{“}fake\text{”}, \text{“}real\text{”}\}$ that are shown in a dataset. Subsequently, we count f the term frequency tf_{dt} and calculate n in every post starting from the first post d_1 as $\sum_{n \in d} f(n, t)$. The tf_{dt} is then collected to form a document matrix $\mathbb{Z}^{m \times k}$ by modifying it into a post-vector representation for the entire document vectors:

$$M_{tf} = \begin{bmatrix} tf_{d_1t_1} & \cdots & tf_{d_1t_k} \\ \vdots & \ddots & \vdots \\ tf_{d_mt_1} & \cdots & tf_{d_mt_k} \end{bmatrix} \quad (1)$$

where $tf_{d_it_j}$ is represented in (2):

$$tf_{d_it_j} = \sum_{n \in d_i} f(n, t_j), \text{where } i \in 1 \ldots m, j \in 1 \ldots k \, . \quad (2)$$

Term weighting or word weighting, $W_{d_it_j}$, aims to assign a weighting value to each term of t_j. Calculation of this weight requires two things, which are $tf_{d_it_j}$(2), the frequency of t_j in the d_i, and the inverse of document frequency, idf_{t_j} (3). The value of idf_{t_j} is inversely proportional to the number of posts containing a specific term t_j. Terms that rarely appear in all posts have an idf_{t_j} value more significant than the idf_{t_j} value of terms that occur frequently. If each post contains a particular term, then the value of the idf_{t_j} term is 0 (zero). These zero values show that the terms appearing in all posts are inadequate to distinguish documents based on specific topics [25]. Using (3) for the term weighting $W_{d_it_j}$of term t_j in media post d_i, we can get (4).

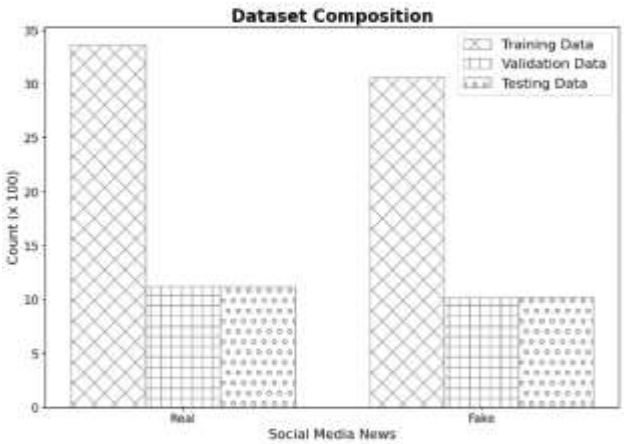

Figure 1. Composition of social media news dataset: real news dominates the training data, while fake news is underrepresented in the testing data

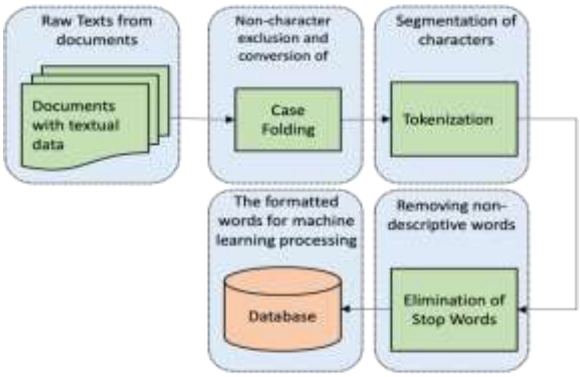

Figure 2. Transforming the unstructured words from different documents into enhanced words to support efficient and accurate classification algorithms

$$idf_{t_j} = \log\left(\frac{m}{f(n,t_j)}\right) \quad (3)$$

$$W_{d_i t_j} = tf_{d_i t_j} \times idf_{t_j} = tf_{d_i t_j} \times \log(\frac{m}{f(n,t_j)}) \quad (4)$$

3.3. Document frequency thresholding

One of the problems commonly found in document classification and clustering processes is the high dimension of the data. Thus, a feature selection is needed to select the most meaningful features that can be used to represent documents. One of the most straightforward feature selection techniques that performs reasonably well is document frequency thresholding (DFT) [11]. DFT is a feature extraction strategy that diminishes the feature space by excluding features with a document frequency (DF) below a preset threshold. For DF, the frequency of occurrence of each distinct term within a training corpus is measured. DF calculates the quantity of documents that include a particular phrase. The calculation of this metric involves tallying the frequency of word occurrence across documents within a collection and subsequently dividing this count by the overall number of documents in the collection. Subsequently, the resultant value can be utilised to determine a threshold for DFT with a computational complexity approximately proportional to the number of training documents. DFT is a commonly employed technique in text categorisation, among other automatic strategies for selection features like mutual information (MI) and information gain (IG) [26].

3.4. Naïve Bayes classifiers

Three methods exist to group documents: supervised, unsupervised, and semi-supervised. In the case of supervised grouping, documents are classified into pre-defined classes and labelled based on the assigned classes. Meanwhile, an unsupervised set categorises an unlabelled dataset based on its hidden features. Semi-supervised is a viable approach for partially labelled data or data with additional outcome measures. One of the central activities in grouping or categorising texts is the supervised approach. The probabilistic MNB classifier-based approach has advantages such as being simple and fast, having high accuracy, and learning to perform reasonably well on relatively small datasets [14]. MNB assigns a given d to the estimated class $\hat{y}$ by (5):

$$\hat{y} = argmax_y \, P(y|d); \quad (5)$$

and $\mathrm{P(y|d)}$ as in (6):

$$P(y|d) = \frac{\left(\prod_{i=1}^{m} P(d_i|y)^{n_i(d)}\right) P(y)}{P(d)}; \; 0 \leq P(d_i) \leq 1 \quad (6)$$

where n is the length of documents. $P(y|d)$ is the posterior probability of target y given the predictor d, or the updated probability for new information. $P(y)$ is the prior probability of y determined by the formula $P(y) = m_y/m$ in which m_y is the number of class y in all posts. $\prod_{i=1}^{m} P(d_i|y)^{n_i(d)}$ is the probability of d_i in the hypothesis given y (likelihood). To prevent zero probabilities, we use the Laplacian smoothing technique for the likelihood [27]. $P(d)$ shows the probability of d (evidence). Since the denominator $P(d)$ is independent of the numerator of (6) means maximizing the posterior probability. Interestingly, MNB compares each of the terms to the statistics of the classes and then predicts the closest class indicated by the highest posterior probability. As MNB is a generative model, the classification performance depends considerably on the underlying training data. Therefore, selecting the suitable hyperparameters of MNB is necessary to get optimal results. In the case of an imbalanced class distribution, where the minority class is underrepresented, the training data for modelling becomes insufficient, leading to biased probability estimates and inaccurate predictions. This imbalance can introduce a bias towards the majority class, reducing sensitivity for the minority class. Considering the impact of imbalanced datasets is crucial to ensure the MNB's independent assumption and improve classification performance. This research also investigated a transformed weight-normalized complement-Naïve Bayes classifier known as CNB [28].

CNB is a modified version of MNB designed to solve text categorisation problems, mainly for imbalanced distribution sets. The standard MNB is changed by CNB, which uses the complement class instead of the target class to determine how likely words appear, a log transformation and a normalisation factor for the word weights, and TFIDF conversion and document length normalisation on the input data. The fundamental purpose of these modifications is to reduce the impact of Naïve Bayes assumptions and improve classifiers' performance. However, some empirical studies have shown that some changes may not be necessary or beneficial for some datasets and that SVM can still outperform CNB on many text categorisation tasks. Furthermore, a study has proposed tweaking the hyperparameter (locally weighted learning) to improve the performance of MNB [29].

3.5. An imbalanced dataset

An imbalanced dataset is a problem often encountered in the classification process in which the used dataset has an unbalanced distribution between each class. The machine learning algorithm automatically assumes that the data are balanced. Based on research conducted by [16], the condition of each class is balanced, 50:50 in both positive and negative classified groups, which obtains 90% accuracy. The errors obtained in the classification process that has been carried out are also evenly distributed among the two groups of data used. As classifiers learn optimally from balanced distributions, the number of instances in each group should be comparable. SMOTE is a derivative of oversampling to make the dataset equally distributed. SMOTE was performed by replicating minority data known as synthetic data points on an imbalanced dataset [30]. The SMOTE method works by looking for k-NN for each data point in the minority class; subsequently, synthetic data points are made as much as the desired percentage of duplication between the minor data points and the k-NN selected based on the hyperparameters. Minority data in an unbalanced dataset will be replicated in synthetic data points based on the results of the best selected k neighbour. Thus, both minority and majority samples has an equal distribution. Therefore, a new balanced dataset can be used for modelling.

4. RESULTS AND DISCUSSION

One of the objectives of this investigation was to compare the performance of different models in the validation and testing process with the previous findings [13] and to propose an algorithm with the lowest runtime. Before conducting the investigation, we followed the original steps and made some adjustments based on the source codes and the investigation routines for only GDBT and DT, as these methods require considerable computation resources. To produce comparable performance, we replicated the reported results as closely as possible. However, we found slight differences as the original studies accommodated randomization. Later, we conducted hyperparameter tuning to improve the replicated performance. To select the best hyperparameter combination and support consistent results in the validation and testing steps, we implemented a greedy cross-validation approach based on GridSearchCV during the training step. Finally, we considered the imbalanced distribution in the dataset by implementing SMOTE. Here, we also use GridSearchCV to select the optimal settings for SMOTE. To support the replicability and reproducibility of this research, we use fixed randomization by implementing a random seed and using the widely recognised libraries, namely scikit-learn and imbalance-learn. The models were evaluated in four metrics: accuracy, precision, recall, and F1 in an Intel(R) Xeon(R) CPU @ 2.30 GHz and 36 GB of RAM. To improve the performance further, we investigated two additional scenarios in the validation and testing steps, specifically hyperparameter tuning and SMOTE integration. Tables 1 and 2 show the results of five models: GDBT, DT, MNB, CNB, and their integrations with SMOTE. The models were also compared based on their runtime, which is the time to complete the validation or testing process in seconds.

Table 1. The performance of models during validation in accuracy (Acc), weighted average measures for precision (Prec), recall (Rec), and F1-score (F1) with the respective runtime (Run) in seconds

Model	Baseline [13]					After hyperparameter tuning					SMOTE and hyperparameter tuning				
	Acc (%)	Prec (%)	Rec (%)	F1 (%)	Run (sec)	Acc (%)	Prec (%)	Rec (%)	F1 (%)	Run (sec)	Acc (%)	Prec (%)	Rec (%)	F1 (%)	Run (sec)
GDBT	86.87	87.12	86.87	86.87	4.38	91.73	91.78	91.73	91.72	52.66	91.78	91.86	91.78	91.77	55.20
DT	85.23	85.26	85.23	85.24	1.10	85.47	85.56	85.47	85.48	1.03	86.12	86.15	86.12	86.13	1.43
MNB	-	-	-	-	-	91.96	91.97	91.96	91.96	0.21	92.20	92.20	92.20	92.20	1.07
CNB	-	-	-	-	-	91.92	91.92	91.92	91.92	0.20	92.20	92.20	92.20	92.20	0.58

Table 2. The performance of models during testing in accuracy (Acc), weighted average measures for precision (Prec), recall (Rec), and F1-score (F1) with the respective runtime (Run) in seconds

Model	Baseline [13]					After hyperparameter tuning					SMOTE and hyperparameter tuning				
	Acc (%)	Prec (%)	Rec (%)	F1 (%)	Run (sec)	Acc (%)	Prec (%)	Rec (%)	F1 (%)	Run (sec)	Acc (%)	Prec (%)	Rec (%)	F1 (%)	Run (sec)
GDBT	86.92	87.20	86.92	86.91	4.29	90.75	90.77	90.75	90.74	52.57	90.70	90.75	90.70	90.70	54.71
DT	85.51	85.55	85.51	85.52	1.08	85.47	85.60	85.47	85.49	1.04	86.54	86.64	86.54	86.56	1.56
MNB	-	-	-	-	-	91.64	91.67	91.64	91.64	0.20	92.10	92.11	92.10	92.11	0.58
CNB	-	-	-	-	-	91.92	91.93	91.92	91.92	0.20	92.10	92.11	92.10	92.11	0.55

As the original investigation did not consider Naïve Bayes classifiers, the performance of MNB and CNB was investigated in only two integrated scenarios: hyperparameter tuning and integration with SMOTE. Despite being previously reported as one of the inferior models, we found that GDBT shows some performance improvements after selecting the appropriate values of parameters, namely the number of estimators, subsample, and learning rate. These parameters were investigated to balance the bias-variance trade-off and prevent overfitting. The results showed considerably improved performance of GDBT during two scenarios in the validation and testing steps. However, the running time for every stage was also significantly longer. On the other hand, DT shows minor improvement even after having hyperparameter tuning and SMOTE integration for the entire process. For the parameters, we tuned the minimum sample split and class weight. The performances were slightly improved after having these parameters tuned. The running time was also generally decreased during the validation and testing steps. Thus, selecting the most appropriate parameters for DT increases the performance and minimises the running time. However, integrating DT with SMOTE showed only minor improvements in the arrangement, with an approximately 40% longer run time.

Lastly, our proposed approach on Naïve Bayes classifiers (MNB and CNB) with hyperparameter tuning and SMOTE integration has the similar performance in terms of accuracy, precision, recall, and F1-score with approximately 92% each. Tables 1 and 2 suggest a trade-off between performance and runtime among the models. After being tuned with SMOTE integration, the CNB still has the fastest runtime, which is the opposite of GDBT. Therefore, the choice of the model depends on the priority of the researcher, whether the highest accuracy or the fastest speed is the preference. However, MNB, and CNB offer promising results with superior performance and short runtime.

5. CONCLUSIONS AND FUTURE WORKS

Based on the research that has been investigated, we can determine that the Naïve Bayes classifiers can be used for identifying fake news about COVID-19 on social media networks, both with and without the SMOTE data balancing algorithm. The MNB and CNB are promising classifiers solving the problem of classifying online fake news about COVID-19 by considering its performance, including accuracy, precision, recall, and F1 values obtained in both validation and testing. After performing hyperparameter tuning, the performance of GDBT significantly improved, while DT showed relatively minimal influence on its results. The performance of the CNB algorithm without involving up-sampling is already superior to the other algorithms, namely DT and GDBT, including MNB itself, in classifying fake news. Although the performance value obtained is slightly lower than that of SVM and LR, the running time results are more promising than the others. Meanwhile, the result values of the CNB method, which includes up-sampling, increase marginally compared to the scenario that does not employ up-sampling. However, the time needed to achieve results in classifying fake news during validation and testing is considerably slower because the up-sampling process using SMOTE takes more time to obtain truly balanced data conditions. Based on the current results, CNB with and without SMOTE can be further improved to make its performance comparable to that of SVM and LR. There are several ways to improve the performance of CNB, with and without SMOTE. However, the exploration of feature engineering, ensemble learning, and data augmentation is still limited. Thus, integrating these methods is potentially of interest for future investigations.

REFERENCES

[1] L. H. Poushy *et al.*, "Satisfaction prediction of online education in COVID-19 situation using data mining techniques: Bangladesh perspective," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 5, p. 5553, Oct. 2022, doi: 10.11591/ijece.v12i5.pp5553-5561.

[2] J. Abbas, D. Wang, Z. Su, and A. Ziapour, "The role of social media in the advent of COVID-19 pandemic: crisis management, mental health challenges and implications," *Risk Management and Healthcare Policy*, vol. Volume 14, pp. 1917–1932, May 2021, doi: 10.2147/RMHP.S284313.

[3] S. Lestari, Y. Yulmaini, A. Aswin, S. Sylvia, Y. A. Pratama, and S. Sulyono, "Implementation of the C4.5 algorithm for micro, small, and medium enterprises classification," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 6, p. 6707, Dec. 2022, doi: 10.11591/ijece.v12i6.pp6707-6715.

[4] D. R. Garfin, R. C. Silver, and E. A. Holman, "The novel coronavirus (COVID-2019) outbreak: amplification of public health consequences by media exposure.," *Health Psychology*, vol. 39, no. 5, pp. 355–357, May 2020, doi: 10.1037/hea0000875.

[5] G. Pennycook, J. McPhetres, Y. Zhang, J. G. Lu, and D. G. Rand, "Fighting COVID-19 misinformation on social media: experimental evidence for a scalable accuracy-nudge intervention," *Psychological Science*, vol. 31, no. 7, pp. 770–780, Jul. 2020, doi: 10.1177/0956797620939054.

[6] N. G. Ramadhan and F. D. Adhinata, "Sentiment analysis on vaccine COVID-19 using word count and Gaussian Naïve Bayes," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 26, no. 3, p. 1765, Jun. 2022, doi: 10.11591/ijeecs.v26.i3.pp1765-1772.

[7] M. S. Islam *et al.*, "COVID-19–related infodemic and its impact on public health: a global social media analysis," *The American Journal of Tropical Medicine and Hygiene*, vol. 103, no. 4, pp. 1621–1629, Oct. 2020, doi: 10.4269/ajtmh.20-0812.

[8] H. F. Alsaif and H. D. Aldossari, "Review of stance detection for rumor verification in social media," *Engineering Applications of Artificial Intelligence*, vol. 119, p. 105801, Mar. 2023, doi: 10.1016/j.engappai.2022.105801.

[9] M. B. Ressan and R. F. Hassan, "Naïve-Bayes family for sentiment analysis during COVID-19 pandemic and classification tweets," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 28, no. 1, p. 375, Oct. 2022, doi: 10.11591/ijeecs.v28.i1.pp375-383.

[10] P. Lin and X. Luo, "A survey of sentiment analysis based on machine learning," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12430 LNAI, 2020, pp. 372–387. doi: 10.1007/978-3-030-60450-9_30.
[11] P. Hajek and R. Henriques, "Mining corporate annual reports for intelligent detection of financial statement fraud – a comparative study of machine learning methods," *Knowledge-Based Systems*, vol. 128, pp. 139–152, Jul. 2017, doi: 10.1016/j.knosys.2017.05.001.
[12] A. Shawon, S. T. Zuhori, F. Mahmud, and M. J.-U. Rahman, "Website Classification using word based multiple N -Gram models and random search oriented feature parameters," in *2018 21st International Conference of Computer and Information Technology (ICCIT)*, IEEE, Dec. 2018, pp. 1–6. doi: 10.1109/ICCITECHN.2018.8631907.
[13] P. Patwa *et al.*, "Fighting an infodemic: COVID-19 fake news dataset," in *Communications in Computer and Information Science*, vol. 1402 CCIS, 2021, pp. 21–29. doi: 10.1007/978-3-030-73696-5_3.
[14] K. Shu, S. Wang, and H. Liu, "Beyond news contents: the role of social context for fake news detection," in *WSDM 2019 - Proceedings of the 12th ACM International Conference on Web Search and Data Mining*, New York, NY, USA: ACM, Jan. 2019, pp. 312–320. doi: 10.1145/3289600.3290994.
[15] F. A. Kwanda and T. T. C. Lin, "Fake news practices in Indonesian newsrooms during and after the Palu earthquake: a hierarchy-of-influences approach," *Information, Communication & Society*, vol. 23, no. 6, pp. 849–866, May 2020, doi: 10.1080/1369118X.2020.1759669.
[16] N. Grinberg, K. Joseph, L. Friedland, B. Swire-Thompson, and D. Lazer, "Political science: fake news on Twitter during the 2016 U.S. presidential election," *Science*, vol. 363, no. 6425, pp. 374–378, Jan. 2019, doi: 10.1126/science.aau2706.
[17] S.-F. Tsao, H. Chen, T. Tisseverasinghe, Y. Yang, L. Li, and Z. A. Butt, "What social media told us in the time of COVID-19: a scoping review," *The Lancet Digital Health*, vol. 3, no. 3, pp. e175–e194, Mar. 2021, doi: 10.1016/S2589-7500(20)30315-0.
[18] R. Das and W. Ahmed, "Rethinking fake news: disinformation and ideology during the time of COVID-19 global pandemic," *IIM Kozhikode Society & Management Review*, vol. 11, no. 1, pp. 146–159, Jan. 2022, doi: 10.1177/22779752211027382.
[19] S. Mishra, R. Kumar, S. K. Tiwari, and P. Ranjan, "Machine learning approaches in the diagnosis of infectious diseases: a review," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 11, no. 6, pp. 3509–3520, Dec. 2022, doi: 10.11591/eei.v11i6.4225.
[20] N. N. Alabid and Z. D. Katheeth, "Sentiment analysis of Twitter posts related to the COVID-19 vaccines," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 24, no. 3, p. 1727, Dec. 2021, doi: 10.11591/ijeecs.v24.i3.pp1727-1734.
[21] F. Fathoni, E. Erwin, and A. Abdiansah, "Multilabel sentiment analysis for classification of the spread of COVID-19 in Indonesia using machine learning," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 31, no. 2, p. 968, Aug. 2023, doi: 10.11591/ijeecs.v31.i2.pp968-978.
[22] S. U. Hassan, J. Ahamed, and K. Ahmad, "Analytics of machine learning-based algorithms for text classification," *Sustainable Operations and Computers*, vol. 3, pp. 238–248, 2022, doi: 10.1016/j.susoc.2022.03.001.
[23] A. Khalid Al-Mashhadany, A. T. Sadiq, S. Mazin Ali, and A. Abbas Ahmed, "Healthcare assessment for beauty centers using hybrid sentiment analysis," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 28, no. 2, p. 890, Nov. 2022, doi: 10.11591/ijeecs.v28.i2.pp890-897.
[24] O. Ajao, A. Garg, and M. Da Costa-Abreu, "Exploring content-based and meta-data analysis for detecting fake news infodemic: a case study on COVID-19," in *2022 12th International Conference on Pattern Recognition Systems (ICPRS)*, IEEE, Jun. 2022, pp. 1–8. doi: 10.1109/ICPRS54038.2022.9854058.
[25] D. M. J. Lazer *et al.*, "The science of fake news," *Science (1979)*, vol. 359, no. 6380, pp. 1094–1096, Mar. 2018, doi: 10.1126/science.aao2998.
[26] Y. Wang, K. Kim, B. Lee, and H. Y. Youn, "Word clustering based on POS feature for efficient twitter sentiment analysis," *Human-centric Computing and Information Sciences*, vol. 8, no. 1, p. 17, Dec. 2018, doi: 10.1186/s13673-018-0140-y.
[27] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," *Computer Speech & Language*, vol. 13, no. 4, pp. 359–393, Oct. 1999, doi: 10.1006/csla.1999.0128.
[28] J. D. M. Rennie, L. Shih, J. Teevan, and D. Karger, "Tackling the poor assumptions of Naive Bayes text classifiers," in *Proceedings, Twentieth International Conference on Machine Learning*, 2003, pp. 616–623. doi: 10.5555/3041838.3041916.
[29] A. M. Kibriya, E. Frank, B. Pfahringer, and G. Holmes, "Multinomial naive bayes for text categorization revisited," in *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, vol. 3339, 2004, pp. 488–499. doi: 10.1007/978-3-540-30549-1_43.
[30] A. Fernandez, S. Garcia, F. Herrera, and N. V. Chawla, "SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary," *Journal of Artificial Intelligence Research*, vol. 61, pp. 863–905, Apr. 2018, doi: 10.1613/jair.1.11192.

BIOGRAPHIES OF AUTHORS

Hasan Dwi Cahyono is a member of the Faculty of Information Technology and Data Science at Universitas Sebelas Maret (UNS) in Surakarta, Central Java, Indonesia. His interests are centered across software engineering, generative modelling, normalizing flow, a crucial aspect that underpins distribution transformations of generative modelling. He can be contacted at email: hasan.dwi.cahyono@gmail.com.

Atara Mahadewa received his B.Sc. in Informatics in 2023 from Universitas Sebelas Maret, Surakarta, Indonesia. His research interests include Artificial intelligence, Natural language processing, and analysis expressed in an undergraduate thesis. He can be contacted at email: ataramahadewa@student.uns.ac.id.

Ardhi Wijayanto is a lecturer and researcher at the Faculty of Information Technology and Data Science, Universitas Sebelas Maret, Surakarta, Indonesia. He finished his master's degree at Universitas Gadjah Mada, Yogyakarta, Indonesia. The topic of his master's degree research is recommendation systems using parallel computation for big data. Now he is working on PhD research at Universiti Tun Husein Onn Malaysia. The topic of his Ph.D. research is the I.C. design of post-quantum cryptography. He also has an interest in open-source software, data mining, and IoT. He can be contacted at email: ardhi.wijayanto@staff.uns.ac.id.

Dewi Wisnu Wardani is a lecturer at the Faculty of Information Technology and Data Science, Universitas Sebelas Maret, Surakarta, Indonesia. She can be contacted at email: dww_ok@uns.ac.id.

Haryono Setiadi is a lecturer at the Faculty of Information Technology and Data Science, Sebelas Maret University, Surakarta, Indonesia. He received his master's degree from Gadjah Mada University, Indonesia in 2009. He is conducting research activities in the areas of educational data mining, software engineering, expert system, and machine learning. He can be contacted at email: hsd@staff.uns.ac.id.